METHOD FOR SECURE TIME-STAMPING OF DIGITAL DOCUMENTS

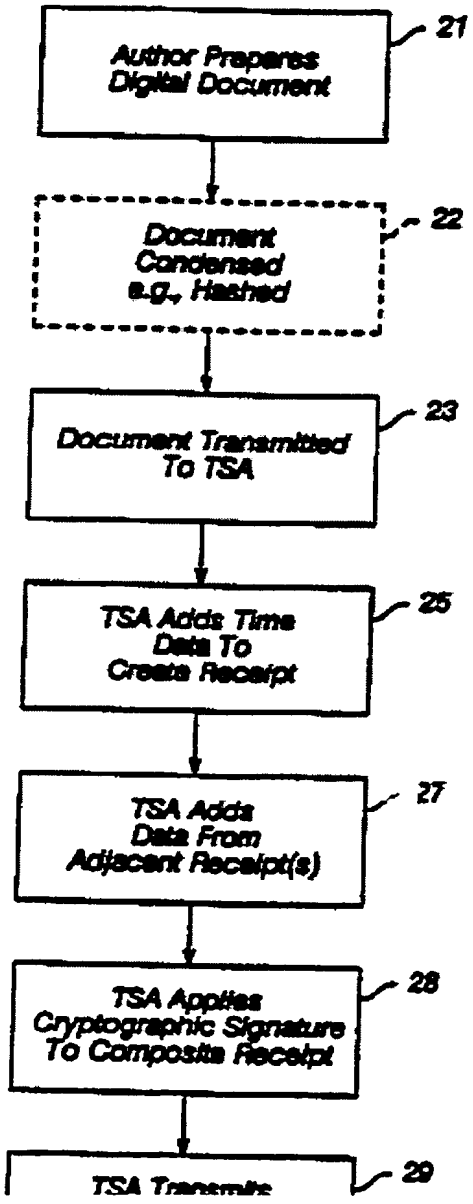
Patent number: WO9203000
Publication date: 1992-02-20
Inventor: HABER STUART ALAN (US); STORNETTA WAKEFIELD SCOTT JR (US)
Applicant: BELL COMMUNICATIONS RES (US)
Classification:
- international: H04L9/00
- european: H04L9/32T
Application number: WO1991US05386 19910730
Priority number(s): US19900561888 19900802; US19910666896 19910308

Also published as:
EP0541727 (A1)
JP2002092220 (A)
EP0541727 (A4)
EP0541727 (B1)

Cited documents:
US4145568
US4206315

Abstract of **WO9203000**

A system for time-stamping a digital document is disclosed which protects the secrecy of the document text and provides a tamper-proof time seal establishing an author's claim to the temporal existence of the document. Initially the author prepares the document (21), which may then be condensed by a process such as hashing (22). Next, the document is transmitted to the Time Stamping Authority (23), which adds time data to create a receipt (25) and data from adjacent receipts (27). Thereafter, the Time Stamping Authority applies a cryptographic signature to the composite receipt (28), which is then transmitted to the author (29).



NTT-1733

(19)日本国特許庁（ＪＰ）　(12)公表特許公報（Ａ）　(11)特許出願公表番号

特表平6－501571

第６部門第２区分　(43)公表日　平成６年(1994)２月17日

(51)Int.Cl.[5]	識別記号	庁内整理番号	ＦＩ		
Ｇ０９Ｃ　1/00		9194－5Ｌ			
Ｈ０４Ｌ　9/32					
		7117－5Ｋ	Ｈ０４Ｌ　9/00		Ａ

審査請求　有　予備審査請求　有　（全 10 頁）

(21)出願番号　特願平3－516026
(86)(22)出願日　平成３年(1991)７月30日
(85)翻訳文提出日　平成５年(1993)２月２日
(86)国際出願番号　ＰＣＴ／ＵＳ９１／０５３８６
(87)国際公開番号　ＷＯ９２／０３０００
(87)国際公開日　平成４年(1992)２月20日
(31)優先権主張番号　５６１，８８８
(32)優先日　1990年８月２日
(33)優先権主張国　米国（ＵＳ）
(31)優先権主張番号　６６６，８９６
(32)優先日　1991年３月８日
(33)優先権主張国　米国（ＵＳ）

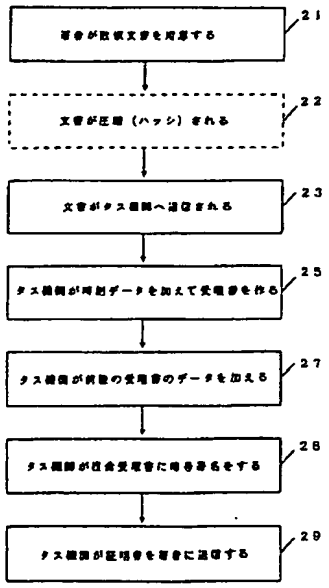
(71)出願人　ベル　コミュニケーションズ　リサーチ　インコーポレーテッド
アメリカ合衆国、07039―2729　ニュージャージー州、リビングストン、ウエスト　マウント　プレザント　アベニュー　290
(72)発明者　ハバー、スチュアート、アラン
アメリカ合衆国、10003　ニューヨーク州、ニューヨーク、アービン　プレイス　22、アパートメント　２シー
(74)代理人　弁理士　小林　孝次

最終頁に続く

(54)【発明の名称】　数値文書にタイムスタンプを確実に押す方法

(57)【要約】

文字数字式やビデオやオーディオや絵のデータを含む、数値文書にタイムスタンプを押すシステムは文書テキストの秘密を守り、その文書が成立した時刻に対する著者の主張を確立する、不正変改の恐れのない時刻のシールを提供します。最初に、文書は一方向性のハッシ関数で一つの数字に圧縮され、これによって文書テキストの独自の表示を確定するかも知れません。本発明の一実施例ではこの数字はそれから外部機関に送信され、そこでその時の時刻が加えられて受理書が作られ、これが公開鍵署名法で機関によって証明されて、文書存在の証拠として著者に返されます。機関によるタイムスタンプに通謀による不正がないようにし、システムの信頼性を高めるために、受理書は他の同じ頃の受理書と結合され、かくして連続の時の流れの中の文書の位置を確定してから、機関によって証明されます。他の実施例では、タイムスタンプされる文書のハッシ数の関数を独自の種として、これによる無作為選択によって複数の機関が指定されます。もう一つの実施例では、機関は受理書のデータにその時の記録連鎖証明書を加えてハッシして受理書を



証明します。ここでその時の記録連鎖証明書は前の受理書の夫々をその時々の連鎖証明書と次々にハッシした結果得られる数です。文書の存在を後で証明するには、機関の公開の鍵を使い、問題の文書の表示を使って証明の段階を繰り返して、証明書の真正であることが認証されます。問題の文書が原文書と同一である時だけ両方の証明書の数が一致します。

特許請求の範囲

1．a）数値文書の数値表示が創作者から外部機関へ通信され、

b）この外部機関がこの数値文書の数値表示の少なくとも一部分とその時の時刻の数値表示とを包含する受理書を作り、

c）この受理書がこの外部機関によって証明できる数値署名署名法によって証明される

ことを特徴とする数値文書にタイムスタンプを能率に押す方法。

2．前記数値文書表示受理書が前記数値文書に決定関数法を適用して得られる数の数値表示の少なくとも一部分を包含する前記特許請求の範囲第1項記載の方法。

3．前記数値数表示が前記数値文書に一方向性ハッシ法を適用して得られる前記特許請求の範囲第2項記載の方法。

4．前記受理書が前記外部機関が受理した他の数値文書の少なくとも一つに特有な時刻表示と数値文書表示を更に包含する前記特許請求の範囲第1項記載の方法。

5．前記外部機関が予め定められた世界から、前記数値文書に決定関数法を適用して得られる数の数値表示の少なくとも一部分を種として疑似無作為発生機で動作為に、選ばれる前記特許請求の範囲第1項記載の方法。

6．前記疑似無作為発生の種が前記数値文書に一方向性ハッシ法を適用して得られる前記特許請求の範囲第5項記載の方法。

7．前記疑似無作為発生によって選ばれた少なくとももう一つの付加的外部機関によっても同様にタイムスタンプ証明書が作られる前記特許請求の範囲第5項記載の方法。

8．前記疑似無作為発生によって選ばれた少なくとももう一つの付加的外部機関によっても同様にタイムスタンプ証明書が作られ、夫々の付加的外部機関の選択時の入力は以前に作られた出力の数値表示に前記一方向性ハッシ法を適用して得られる出力の数値表示の少なくとも一部分である。前記特許請求の範囲第7項記載の方法。

9．a）一つのシリーズの文書の特定の一つの数値表示を作り、

b）前記特定文書表示と前記シリーズ中の前記特定文書の直前の文書に対する証明書記載連鎖表示を包含する連鎖に対して決定関数法を適用して前記特定文書に対する証明書記載連鎖表示を作る

ことを特徴とする一つのシリーズの数値文書の時間的順序を証明する方法。

10．前記シリーズの以後の文書の夫々に対して前記の段階を繰り返すことを更に包含する前記特許請求の範囲第9項記載の方法。

11．前記文書表示の夫々が前記文書に決定関数法を適用して得られる前記特許請求の範囲第10項記載の方法。

12．数値文書の数値表示を外部機関に通信し、前記外部機関がこの時の時刻の数値表示と前記数値文書の数値表示の少なくとも一部分を包含する受理書を作り、前記外部機関で前記受理書を証明する時、

a）前記受理書の数値表示を以前の証明書記載連鎖の表示と連鎖して複合表示を作り、

b）前記複合表示に決定関数法を適用して前記受理書に対する証明書記載連鎖表示を作る

ことによって前記受理書を証明することを特徴とする数値文書にタイムスタンプを押す方法。

13．前記外部機関がこれらのタイムスタンプ処理の証明書記載連鎖表を包含する記録を維持する前記特許請求の範囲第12項記載の数値文書にタイムスタンプを押す方法。

14．前記受理書に含まれる数値文書表示が前記数値文書に決定関数法を適用して得られる他の数値表示の少なくとも一部分を包含する前記特許請求の範囲第12項記載の数値文書にタイムスタンプを押す方法。

15．前記数値表示が前記数値文書に一方向性ハッシ法を適用して得られる前記特許請求の範囲第14項記載の数値文書にタイムスタンプを押す方法。

明細書

数値文書にタイムスタンプを確実に押す方法

発明の背景

文書が書かれた日付を立証し、問題の文書の内容が日付の押された原文書の内容と実質に同じであることを証明することが多くの場合に必要です。例えば、知的財産に関しては、ある人が発明の内容を最初に記載した日付を実証することは極々決定的です。発明の考えをタイムスタンプする普通のやり方は、研究室の記録帳に自分の仕事を毎日書込むことです。消せないようには付を書いて署名した記載が記録帳の各ページに次々と書込まれ、頁番号を打たれて綴込まれたページは記録を消らないように必要することを期待にします。記事の正当性は、一般に利害関係のない第三者によって定期的に検閲され証人として署名されることによって、更に高められます。何時考えたかということが後で証明されなければいけなくなった時、記録帳の物理的な内容と定められた記録の手順の両方が、少なくとも記録帳の証人の日付の時には考えが存在していたという事実を実証する効果的な証拠となります。

読むことのできるテキストの数値的な表示だけでなく、ビデオやオーディオや絵のデータをも含む、電子文書が際々と広く使われるようになって来て、このような文書の日付を確立する「記録帳」の概念の実行可能性が考かされています。電子数値文書は極めて容易に改訂され、このような改訂は後に証拠を残さないので、ある文書が作られた日付を本当にその文書が示しているのか、又元来のメッセージを今でも本当に表しているのかについて

は、信頼できる証拠は得られています。同じ理由で、実証する署名の信頼性についても重大な疑いが起きて来ます。数値文書の内容の改訂を許さない効果的な手順がないと、システムの信頼性が基本的に欠けていることは電子文書の有効性がもっと広く採用されることを妨げます。

現在でも、電子文書の通信を確証する若干の手順があります。しかし、実際にはこのような手順は両方向の通信に限られます。即ち、このような通信では、通信者は通信される文書の元来の内容と発信者を受信者に立証しようと本質的に望みます。例えば、「秘密の鍵」を使う暗号法は長い間、限られた数の、お互いに知合っていて暗号を解く鍵を知っている個人の間で、メッセージの通信に使われてきました。メッセージを暗号にすることは不正変更を防ぎ、秘密の鍵を使うと通信されたメッセージの「平文」が得られると言う事実が、メッセージは決まったグループの一員が通信したものである証拠となります。しかし、メッセージを書いた時刻は間接的に、受信者が受取った時刻より後ではないと、証明されるに過ぎません。それで、この方法は限られない世界で後になって役に立つタイムスタンプの証拠を提供しません。

もっと広く適用される実証通信法、即ち「公開の鍵」を使う暗号法が、ディッフィーとヘルマン（「暗号法の新しい方向」、IEEE 情報理論雑誌、第IT－22号、昭和51年11月、644－654ページ）によって記述され、その後リベスト等によって、昭和58年9月20日付のアメリカ合衆国特許4，405，829号で実行されました。この方法は利用者の世界を、公表された名簿以外ではお互いに未知の、実質上限定されない数のシステム加入者に拡大しましたが、実証できる通信は依然として両方向のものでした。通信者の秘密の鍵で暗号化されたメッセージの公開の鍵での解読を伴うもののような、公開の鍵の「署名」は、限定されない世界のどのメンバーにもメッセージの送信者が確かに

ついて確実な証拠を提供しますが、このメッセージの受取人だけが、メッセージは受取った時刻以前に存在したことを知ることができますから、この限界は今でもあります。しかし、このような受取はメッセージが存在した時刻の直後の証拠を全世界に提供はしません。受取ったメッセージに関連する受取人の証言はメッセージの内容とその存在の時刻についての証拠を提供しますが、このような証拠は電子数値文書の内容が、送信者または証人によって簡単に改変できるという基本的な問題を抱えています。

従って、総ての文書が簡単に改変できる数値形式で書かれる世界になるという予想は、このような文書の信頼性を確立する既存の手順を本質的に変うくします。数値文書の内容と時刻を確定し、少なくとも有形文書の場合に現在認められている程度に、内容と時刻に関して直接的な証拠を提供することができるような実証のシステムが現在明白に緊要とされています。

発明の概要

この発明は数値文書をタイムスタンプする方法において信頼できるシステムを作り、現在の記録帳の本質的な特徴の二つと同等のものを提供します。第一に、文書の内容とその存在のタイムスタンプは、文書の数値データに消えないように組込まれ、これによって出来たタイムスタンプされたデータのいかなる部分も、改変が明白とならないように改変することは不可能であります。このように、文書のテキストの状態はタイムスタンプの瞬間に確定されます。第二に、数値文書がスタンプされた時刻は、虚偽の時刻の長期を組込むことを防ぐ、数値的に「証人として」署名する手順で確認されます。基本的に、この方法はタイムスタンプの処理のコントロールを署者から独立機関へと移し、真の時刻以外のスタンプをするよう機関に影響を及ぼす能力を署者から取上げます。

この発明の方法は、文書の署者が通信網の中に沢山散らばっていると仮定します。このような署者は個人、会社、会社の部門等で、夫々が区別され、識別番号等で特定できる、署者世界の一員です。この発明の一つの実施例では、この世界はタイムスタンプ機関（タス機関）の数個人で構成されます。もう一つの実施例では、散らばった署者の夫々がこの世界の他のメンバーのために タイムスタンプのサービスを行う機関であります。

一般の適用においては、図面の第1図に示されるように、この方法では、署者が広く文字、数字、音声、画像の表示を包含する数値文書を準備し、この文書を、好ましくは圧縮した形で、タス機関へ送信します。タス機関は受理した時刻を表す数値データを加えて文書にタイムスタンプし、この文書にその機関の署名を入れて暗号化、できた文書即ち原文書の存在時刻証明書を署者に返信し、署者はこのような存在を証明することが必要になる時のために保管します。他の方法では、タス機関は受理した時刻を表す数値データを加えて文書にタイムスタンプし、受理書を作ります。これまでの受理書を暗号返信したものにこの受理書を連鎖し、この結合文書から以下に説明する決定関数を使って新しい連鎖文書を作ります。これによってできた連鎖値を時刻その他の認識データと一緒にして証明書を作ります。

タス機関への送信中に秘密文書の情報が盗聴されるのを防ぐために、また全文書の通信に関する数値容量値を減らすために、署者は場合によっては数値文書の一部分を決定関数を使って数値のサイズを大幅に圧縮して独自の値に変換するかも知れません。決定関数としては、例えば専門分野では「一方向性ハッシュ関数」として知られる多数のアルゴリズムのどの一つでも使えます。ハッシュ関数のこのような応用は、例えばダムガードによって文書

署名技法における安全な改良の基礎の中で述べられています（「衝突のないハッシュ関数と公開の鍵を使う署名法」、暗号学の進歩--ユーロクリプト１９８７、スプリンガー・フェルラーク、ＬＮＣＳ、１９８８、第３０４巻、２０３－２１７ページ）。しかし、この発明の応用では、ハッシング法に典型的な「一方向」性はもう一つの目的に叶います。すなわち、タス機関がタイムスタンプを押したり、文書を連鎖証明書に組込んだ後では、文書は密かに変改されることはできないという保証を提供します。

ハッシング関数は丁度このような保証を提供します。というのは、署名の原署や合成連鎖受理書のような文書がハッシされる時に元の内容の代表的な「指紋」が作られ、これから元の文書を復元することは、ほとんど不可能です。それゆえに、タイムスタンプされた文書は署名の後によって改変されることは不可能です。署名もまた発行されたタイムスタンプ証明書を文書の改訂版に適用することはできません。なぜならば、原文書の内容の変化は、たとえ一語または数値データの一ビットでも、違った文書となり、全く違った指紋値のものにハッシするからです。代表的なハッシ値から文書を復元することはできませんが、それにもかかわらず、原文書と主張されているものはこのタイムスタンプ手順で証明されます。というのは原文書表示の真のコピーを包含する受理書は、元のハッシング法を使えば署名の持っている証明書に書かれている、元の数字または同じ連鎖値に何時でもハッシするという事実があるからです。

この手順では現在あるどんな決定関数でも使えますが、たとえば、リベスト（「ＭＤ４」メッセージ・ダイジェスト・アルゴリズム」、暗号学の進歩--クリプト１９９０、スプリンガー・フェルラーク、ＬＮＣＳ、近刊予定）が述べているような一方向性ハッシ関数を引用してここに組入れて置きます。この発明の実用においては、かようなハッシング操作は場合によっては署名によって通信中の期限という苦しい利点のためになされます。文書

が暗号文でない形で受理された場合にはタス機関がハッシングするかも知れません。文書の内容と組込んだ時刻のデータが改変されないようにどのように確定されても、このシステムの信頼性を増すためには、未定世界のメンバーに対して、受理書は、署名ではなく、実際にタス機関によって作られ、示された時刻は正しく、例えば署名と共謀したタス機関が非公式に公表したものではないと証明する使用が残っています。

第一の問題に対しては、タス機関は、前述の公開の鍵の方法のような、実証できる署名法を用いて、署名へ通信する前にタイムスタンプを押したと証明します。後で、タス機関の公開の鍵での原鍵での署名の確認は、署名と世界全体に対して、証明書はタス機関が作ったものであると証明します。しかしながら、タイムスタンプ自身の真実性の証明は、以下に述べるこの発明の他の部分に依存します。

別の方法では、タス機関は、新しく受理したものを一つ一つその時までの連鎖に付け加え、この複合表示に決定関数を適用し、即ちハッシングを行い、新しい連鎖を作って、順次にタイムスタンプした処理の記録を維持します。この連鎖はハッシング連鎖によって作られた鎖で、これが署名に与えられる受理書または証明書に記されて、そこに示されるタイムスタンプを証明するのに役立ちます。後で証明書の確認をするのには、署名の時刻受理書とタス機関の記録にあるその証明の連鎖の値の組合わせに再度ハッシを行います。その結果署名の証明書に記載の連鎖値が出れば、署名と全世界に対してその証明書はタス機関で作られたものであると証明します。この結果はまたタイムスタンプの真実性をも証明します。というのは元の受理書に記載の後での元の表示を使わなければ、ハッシング関数によって元の証明書に記載の連鎖値を作ることはできないからです。

第２図に一般的に書かれているような、この手順の一つの実施例では、署名の世界からタ

ス機関の施設へと比較的に直線な文書の流れを利用します。夫々の処理した文書D_kに対してタス機関はタイムスタンプ受理書を発行し、これには、たとえば、連鎖受理番号r_k、署名A_kの原鍵番号ＩＤ$_k$等による証明、文書のハッシH_k、その時の時刻t_kが含まれます。タス機関はこの他に、直前に処理した署名A_{k-1}の文書D_{k-1}の受理データも含め、これによって文書D_kのタイムスタンプは独自に確立された前の受理時刻t_{k-1}によって「過去」の方向は限定されます。同様に、次に受理した文書D_{k+1}の受理データも、文書D_kのタイムスタンプを「将来」の方向に限定するために、含められます。複合受理書は今や３つ、あるいは希望によってはそれ以上の、連鎖したタイムスタンプ受理書の時刻のデータを含み、あるいはそれらの原鍵部分を含み、タス機関の暗号署名で証明されて、署名A_kに送信されます。同様にして、D_kとD_{k+2}の原鍵表示を含む証明書が署名A_{k+1}に送信されます。このようにして、タス機関によって出されたタイムスタンプ証明書の夫々は連鎖した時間の中で確定され、記付された多数の連続した証明書を照合すれば順番が違っていれば直ちに判るので、タス機関はどれも偽って発行することはできません。時の流れでの文書のこのような順次の確定は非常に効果的なので、タス機関の署名は実際には不必要かもしれません。

第３図に一般的に書かれているような、この手順の第二の実施例では、たとえばタイムスタンプの手順を利用する多数の署名といった、広い世界の中にタイムスタンプの仕事を独作為に配付します。タス機関を管理の目的に使ってもよく、あるいは依頼する署名は直接選択したタイムスタンプする署名並機関と連絡してもよいわけです。いづれにしても、署名とタス機関の共謀でタイムスタンプが文書に押されたのではないという保証が上記の様に必要で、これは少なくとも機関の世界のある部分は変造しようとする署名に責任されないか、そのような署名に悪質の合成をもたらすという合理的な結論と、特定の文書をタイ

ムスタンプする機関はこの世界から全く独作為に選ばれるという事実の両方で満たされます。署名が署名の自身の選択で共謀しそうな機関を選ぶことが出来ないことは、原理的な時刻の偽造の可能性を事実上除きます。

この世界の個人のメンバーの中から予定数の機関を選ぶのは、インバグリアッツオ、レビンとルビー（「一方向性関数による責任独作為発生」、第２１回ＳＴＯＣ議事録、１２－２４ページ、ＡＣＭ、１９８９）によって論じられた型の責任独作為発生機によってです。これに対する最初の種はタイムスタンプされる文書の、ハッシのような、決定関数であります。種の入力として文書のハッシや他のこのような関数を与えられると、条件を満たす責任独作為発生機は一群の機関の原鍵番号を出力します。この機関の選択は実際上予測できず独作為です。

機関が選ばれると、タイムスタンプは前述のように行われますが、夫々の機関は個性的に受理時刻のデータを受理した文書に付け加え、その結果できたタイムスタンプした別の受理書を機関固有の証明可能の暗号署名で証明し、証明書を署名に通信します。この通信は申請した署名に直接の場合もあり、管理するタス機関を経由する場合もあり、後者の場合にはタス機関が更に証明を付け加えるかも知れません。署名をするという機関と公表された署名の認識番号表は、実際に責任独作為発生機で選択された機関を利用したことの証明を与えます。本発明の分布した機関を使う実施例は受理書を連鎖する方法に比べて、タイムスタンプ証明書がより早く発行され、また文書の署名の後での証明は他の署名の証明書が入手できるかどうかに余り依存しない利点があります。

第４図に示される別の実施例では、タス機関が作るタイムスタンプ受理書に、たとえば受理処理順を番号r_k、署名の原鍵、たとえば原鍵番号ＩＤ$_k$等、文書の数値表示、たとえばハッシH_k、とその時刻t_kを含めます。この後タス機関は受理書のこれらのデータ（また

はその代表的な任意の部分）を、その直後に処理した、署名A_{u-1}の文書D_{u-1}の証明書記載基礎値C_{u-1}に含まし、これによって文書D_uのタイムスタンプを、独自に確立された前の受理時刻t_{u-1}で限定します。

この複合データの数列（r_u，ID_u，H_u，t_u，C_{u-1}）にその後ハッシされて新しい基礎値C_uとなり、これが処理番号r_uとともにタス機関の記録に入れられ、またタイムスタンプ受理書データとともに証明書記載基礎値としてA_uに送信されます。同様にして、C_uと書類D_{u+1}の受理書のタイムスタンプ要素をハッシして得られる証明書値が署名A_{u+1}に送信されます。このようにして、タス機関が出したタイムスタンプを押した基礎証明書の文々は連鎖した時の中に確定され、タス機関は偽って作ることは出来ません。何故ならば前の証明書とハッシして証明書記載基礎値を再生しようとすれば矛盾を示すからです。

第5図に示されるような、この発明のより一般的な適用においては、特定の文書の表示、すなわちハッシは連鎖の文書の証明書記載基礎値と単に連結され、この複合表示の決定関数表示、たとえばやはりハッシ、が次に作られて、この特定の文書の記録上の基礎値として維持されます。この増大して行くシリーズの以後の夫々の文書は同様に処理されて記録を拡張し、この記録自身がこのシリーズの中で、もっと広く見れば連続した時の中で、このような文書の夫々が占める位置の信頼できる証明となります。本発明のこの実施例は、たとえば組織がその活動上の数値の文書や記録の順番や連続性を広くに証明できる信頼できる方法を提供します。

本発明の手順の別の変化では、署名の連鎖の中である時間の内に、これは活動の程度によりますがたとえば一日とかそれ以上の間に、作られた（好ましくはハッシしたりその他の表示の形の）文書の集積をハッシして、タイムスタンプと証明に好都合な単一の文書とします。また、最低許容発生器の最初の値は、その文書によるだけでなく、時記の関数や前に受理書が出された文書にもよるかもしれません。別の方法では、一つの組織のなかで信ぜられた人が、定数する「外部の」機関として、この手順を使ってその組織の文書の連鎖証明書の記録を維持し、定期的にその時々の連鎖証明書をタス機関に送信します。このようにして、ある組織の活動上の記録の順番が、組織の中でも、また外部的にはタス機関を通じて、確立されます。

また、手順実施例の実行は、区文書表示の受信・ハッシング・連鎖、タイムスタンプ押印、証明書記載連鎖値の計算と記録、受理証明書の発行という諸段階を区別行う、単一の電算機のプログラムで直ちに自動化されます。

図面

本発明の説明には以下の図面を用います：

第1図は本発明による文書タイムスタンプの一般手順の流れ図です。

第2図はこの手順の特定の実施例の流れ図です。

第3図はこの手順のもう一つの特定の実施例の流れ図です。

第4図はタイムスタンプ手順の他の実施例の流れ図です。

第5図は本発明による一般連鎖手順の流れ図です。

発明の記述

本発明の実施例を適用した以下の説例で、含まれた手順を更に説明します。説明の便宜上、選ばれた決定関数は上記のリベストによって記述された md4 ハッシング法で、また証明できる署名法はディフィーとヘルマンによって示唆されリベスト等によってアメリカ合衆国特許4，405，829号で実行された公開の鍵の方法です。タス機関が実際に選ぶ関数は色々な手に入る算法の中のどれでも良いのです。どのような算法が用いられても、何をどの順序使ったかという記録は、受理証明書を後で確認するために維持されなければなりません。更に、手順の説明を簡単にするためと以下に述べるそれ以外の理由のために、数字の代表的な部分だけを用います。

第2図に示される本発明の受理書連鎖の実施例を最初に考えましょう。この手順はどの様な長さの文書にも使えますが、以下の適切な引用は、ある署名が段階21で書いてタイムスタンプを希望する文書D_uを充分に代表するものです。

```
Time's glory is to calm contending kings,
To unmask falsehood, and bring truth to light,
To stamp the seal of time in aged things,
To wake the morn, and sentinel the night,
To wrong the wronger till he render right;

        The Rape of Lecrece
```

破線で囲まれた任意段階22で、この文書はmd4算法によって都合の128ビットの数H_uにハッシされますが、この数H_uは16進法では

e f 6 d f d c d 8 3 3 f 3 a 4 3 d 4 5 1 5 a 9 f b 5 c e 3 9 1 5

となります。1000人からなる署名世界の中でシステム記識番号ID_uが172である署名A_uがこの証識番号を付けた文書を段階23でメッセージ（ID_u，H_u）：

1 7 2，e f 6 d f d c d 8 3 3 f 3 a 4 3 d 4 5 1 5 a 9 f b 5 c e 3 9 1 5

としてシステムのタス機関に、この文書をタイムスタンプする要請として、送信します。

タス機関は、段階25で、たとえば132という受理書証を番号r_uと、その時の時刻t_uの陳述を付け加えて、文書D_uの受理書を発行します。この時刻の陳述は、署名A_uができたタイムスタンプ証明書を容易に訴めるようにするために、電算機の時計の時刻の精確32ビット表示と文章による供述を、たとえば1990年3月10日グリニジ平均時16：37：41のように含めるかもしれません。そうすると受理書は数列（r_u，t_u，ID_u，H_u）を包含します。

この点で、表示セグメントの数のサイズを前述のように縮らすということを更に考えることが妥当であります。リベスト等によってアメリカ合衆国特許4，405，829号で記述されたように、この例で使われる暗号公開鍵法（この分野では一般に「RSA」署名法として知られています）は、長いメッセージを、一つ一つが暗号化鍵数要素nを越えない数で表されるブロックに分割することが必要です。夫々のこのブロックはこの後RSA法で署名され、送信された後ふたたびアセンブルされます。それゆえに、RSA法で証明する最終の受理書数列が単一のブロックであることを維持しながら、この例で妥当な大きさの数nを使えるためには、受理書数列の夫々の要素は代表的な8ビットに縮らされますが、長すぎる数列の場合には普通は最後の8ビットとなり、このビットは16進法では2つのヘクサデシマルの字となります。それで、たとえば、128ビットの文書ハッシHは最後の8ビット、すなわち0001　0101で表され、これは16進法では15と書かれます。同様にして、ID_uの172は1010　1100で、16進法ではａｃとなりま

す。実際の計算を行わないで、時刻表示t_nは51と表示されると仮定しましょう。受理番号132は84と表示されます。この点で受理書の数列（r_n, t_n, ID_n, H_n）は

8451ac15となりました。

ここで、直前の文書D_{n-1}はタス機関によって1990年3月10日16：32：30に（t_{n-1}の表示は64）に申請

201，d2d67232a61d616f7b87dc145c575174

として処理されたと仮定しましょう。段階27でタス機関はこれらのデータをD_nに対する受理書数列に加えて、16進数の表示、8451ac1584c974、を作ります。この受理書R_nは今やD_nに対する時刻と、それは時には署名A_nがD_nが存在したと主張できない時刻t_{n-1}を確定するデータを含みます。A_nに対するこの確定は、前の署名A_{n-1}が時刻証明書c_{n-1}を保持し、それがt_{n-1}は署名A_{n-2}の証明書にあるリンクされた時刻のデータt_{n-2}の以後であると確定し、というように、証明が必要なだけ続くからです。

タス機関が文書D_nの受理書を実際に発行したことを確立するために、段階28でタス機関は公開鍵暗号署名法で署名をし、段階29でこの受理書は署名A_nに送信されて受理証明書または証明書c_nとなります。上のようにして得られたデータを使い、またタス機関に十進法でRSA署名鍵セット

<n，e>=<432006778214281O9,191>（公開）

<n，d>=<432006778214281O9,2940360242249791>（秘密）

を持つとすれば、R_n、8451ac1564c974、に対する署名付き証明書は

$R^e \bmod n = 39894704664774392$

と計算されるでしょう。署名A_nがこの証明書c_nとR_nの文書のステートメントを受取った時、タス機関の公開の鍵を適用すると

$c_n^e \bmod n = R_n$

となることから、R_nは実際に文書のハッシュH_nを表示するデータを含んでいると確証され、c_nが正確であると迅速に確認されます。

この簡単な1リンクの例の手順で作られた証明書は文書D_nのデータで時間を限定されるので、署名A_{n-1}に対して、文書D_{n-1}は文書D_nの存在のかなり前に時間を遡らせたのではないという信頼できる証拠を提供します。A_nの証明書が以後に処理された文書D_{n-1}からのデータを加えて拡大された時、この証明書は同様に効果的に限定され、A_nが主張するタイムスタンプを立証します。同じ効果を得る別法としては、A_nにA_{n-1}の名を教え、A_nはその署名から1リンク証明書c_{n-1}が要求H_nを含むことを確認できます。この手順は変化させて、任意の数の署名のデータを含む受理証明書を発行するようにすることもでき、追加する毎に変更がないという保証の度合いが高まります。

第3図に示される本発明の別の実施例は署名世界の中から無作為に選ばれたメンバーがタス機関（または証人）となり、すなわち「分布信託」の手順ですが、これは以下のように行われます。実際の運用ではこれらの数はそんなには限定されないのですが、この例では、世界は1000人の署名を含み、そのIDは0ないし999で、タイムスタンプの真実性を確立するのには3人の証人がいれば充分と仮定しましょう。また、この例ではタス機関のサービスを含める前記の変化が実行されています。前の例で用いられたハッシング関数、md4、がここでも、任意の段階32で、署名世界から3人の証人を無作為に選択する様をまく決定文書関数の一例として用いられています。

前例の時と同じく、署名は文書をタス機関へ、普通ハッシュした形で、部署番号を付けた申請として送信します：

172，ef6dfdcd833f3a43d4515a9fb5ce3915

タス機関は、段階33で、この文書ハッシュ数列を最初の証人の部署番号を作る種として用い、段階35で、選択法

ID = [md4 (種)] mod (世界の大きさ)

によって選びます。作られた値ハッシュ：

26f54eee92511dbb5e06e7c2de6e0fcf

は128ビットの数を表し、そのmod 1000が487で、これが最初に選ばれた証人のIDです。次の証人も同様にして選ばれ、この後のハッシュ表示を第2の選択の計算に使って

882653ee04d16b1f0d604883ae27300b

を得ますが、このmod 1000は571で、これが第2の証人のIDです。この計算を繰り返し、前の値のハッシュを種に使って最後の証人を596として選びますが、これは

2fe8768ef3532f15c40acf1341902c1e mod 1000

です。

段階37で、タス機関は最初の申請書の写しをこれら3人の証人のそれぞれに送り、段階38で、証人は各個にその時の時刻のステートメントとIDを加え、こうしてできた受理書にRSA暗号署名法で署名して証明し、段階39で証明書を直接署名にまたはタス機関を通じて送信します。後の場合には、タス機関は証明書を一つのファイルにアセンブルして署名に送るかも知れません。証人の選択に当たって確率無作為発生を使うことは個人的な選択を防ぐという事実のために、署名は非協力的な証人がタイムスタンプ証明の前に虚偽の時刻の記入を計画するために通謀しようと試みるのに出あうという危険を避けられます。手順の別法として、署名が直接証人に申請することが許される場合、問題の文書自身が本質的に鍵となる証人の無作為選択により、署名が文書を知人で協力的な証人に向けようとする試みを難しくします。できた一群の証明書は、前述のように署名確認をして、安心して後の証明に使えます。

図面第4図の段階41のように、タイムスタンプ手順での連鎖証明書の作成は、署名A_nが数値文書を獲得することから始ります。前述のように、この数値文書は文字数字式テキスト、ビデオ、オーディオ、絵または確定したデータの他の形のものの数値的な表または表示であるかもしれません。この手順はどのような長さの文書に対しても用いられますが、以下の引用はタイムスタンプしたい文書D_nを充分に代表します：

...the idea in which affirmation of the world and ethics are contained side by side ... the ethical acceptance of the world and of life, together with the ideals of civilization contained in this concept ... truth has no special time of its own. Its hour is now -- always.

Schweitzer

署名が希望すれば、文書D_nは安全と通信に必要な帯域幅を減らすために、例えばmd4法で圧縮されます。段階で図された任意の段階42で示されるように、文書は種類の128ビットの形の値H_nにハッシュされます。これは16進数で

ec2ef3ea60df10cb621c4fb3f8dc34c7

となります。この点で注釈しておきますが、この例で用いられるSHA法やその他の数値表示は本発明の実施に決定的ではありません。すなわち、考えられた手順によって選ばれたこれらの値のどの部分もまたは他の表示も同様に作用します。

1000人の署名世界の中で認識番号ID_Aが634である署名A_Aが、段階43でシステムのタス機関に、以下の署名メッセージ（ID_A, H_D）で、文書にタイムスタンプを押すよう要求し、文書を送信します：

634, ee2ef3ea60df10cb621c4fb3f8dc34c7

段階44で、タス機関は、受理処理数を番号r_n、例えば1328、とその時の時刻t_nの表示を加えて文書D_Aの受理書を作ります。この時刻の表示は電算機の時計の時刻の標準2進表示かも知れず、または最終的なタイムスタンプ証明書が容易に読めるように、単に文章の表示で、例えば1991年3月6日グリニジ平均時19：46：28であるかも知れません。この時、受理書は数列（r_n, t_n, ID_A, H_D）を包含し、これは

1328, 194628GMT06MAR91, 634,
ee2ef3ea60df10cb621c4fb3f8dc34c7

となります。

本発明によれば、この時のタス機関の記録は、例えば、その時の記録逓順と夫々の受理を次々とハッシしてできた値のBで、以前の受理処理時での逓順を含みます。かくして、この逓順記録は以下のようにしてできたものです。最初の処理（$r_n=1$）では受理書は初期値、すなわちタス機関の記憶のハッシと共にハッシされて最初の逓順値c_1を作り、これが最初の処理の証明書の値として使われます。次の処理では、受理書はc_1と逓順され、

それがハッシされて第2の証明書記載逓順値c_2を作り、タス機関のタイムスタンプ署名の全歴史を通じてこれが続きます。

現在の例の直前に文書D_{n-1}がタス機関によって、第1327番目の受理書として処理されて、証明書記載逓順値c_{n-1}

26f54eaa92516b1f0d6047c2de6e0fcf

を作ったと仮定しましょう。手順の段階45で、タス機関はこの値とD_nの受理書を逓順して

26f54eaa92516b1f0d6047c2de6e0fcf,
1328, 194628GMT06MAR91, 634,
ee2ef3ea60df10cb621c4fb3f8dc34c7

を作ります。この複合表示が、段階46で、タス機関にハッシされて、新しい証明書記載逓順値c_nとして

46f7d75f0fbea95e96fc38472aa28ca1

を作ります。

この後タス機関はこの値をその記録に加えて、段階47で署名A_nにタイムスタンプ証明書を送信します。これには以下の証明書記載逓順値もよくまれます：

処理番号：　1328
依頼人認識番号：634
時刻：　19：46：28グリニジ平均時

日付：　1991年3月6日
証明書数：　46f7d75f0fbea95e96fc38472aa28ca1

この手順はタス機関によって以後のタイムスタンプ要求の都度繰り返されます。A_{n-1}からの次の要求がハッシされた形H_{n-1}の文書

201, 882653ee04d511dbb5e06883aa27300b

で1991年3月6日グリニジ平均時19：57：52に受理されたとすると、複合逓順は

46f7d75f0fbea95e96fc38472aa28ca1,
1329, 195752GMT06MAR1991, 201,
882653ee04d511dbb5e06883aa27300b

となり、A_{n-1}に返信される証明書は

処理番号：　1329
依頼人認識番号：201
時刻：　19：57：52グリニジ平均時
日付：　1991年3月6日
証明書数：　d9bb1b11d58bb09c2763e7915fbb83ad

となります。

将来、署名A_{n-1}が文書D_{n-1}はタス機関によって1991年3月6日19：57：52に受理されたと証明しようと望むならば、タス機関の記録が調べられ、直前に処理された1328の逓順受理書値c_n：

46f7d75f0fbea95e96fc38472aa28ca1

が得られます。証明しようとする文書はタス機関に送信された時の形、即ちハッシに変換され、この値がc_nやその他のA_{n-1}の証明書に記載のデータと逓順されます。問題の文書が本物であれば、複合表示は

46f7d75f0fbea95e96fc38472aa28ca1,
1329, 195752GMT06MAR1991, 201,
882653ee04d511dbb5e06883aa27300b

となり、これをハッシすると正しい証明書記載逓順値

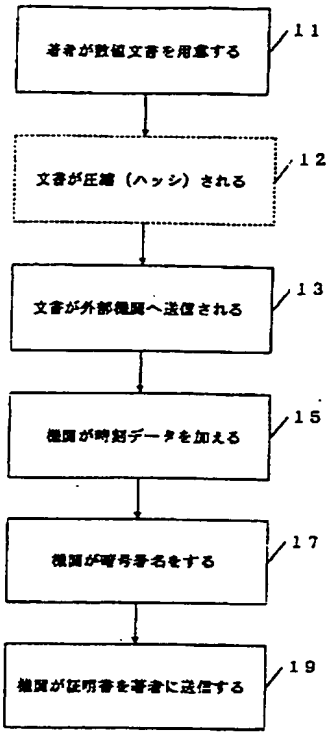
d9bb1b11d58bb09c2763e7915fbb83ad

となって、問題の文書はD_{n-1}であることが証明されます。さもなければ、改訂された文書はハッシされると違った値になり、これを要素として含む複合表示をハッシしたものは、処理番号1329の証明書に記載の値と違った証明書記載逓順値となります。

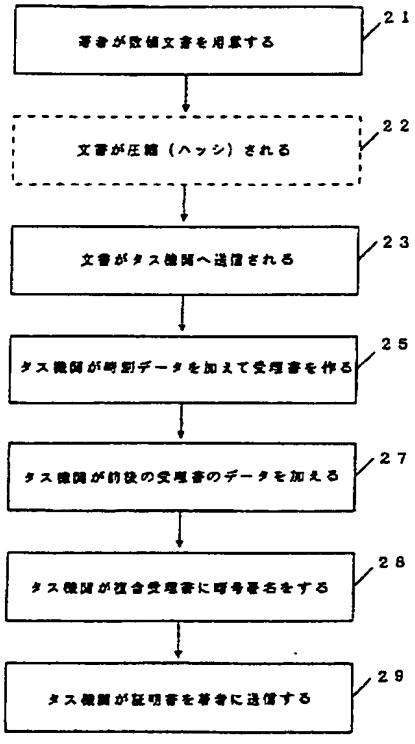
もしもっと証明が必要ならば、例えば文書を改変した後でc_{n-1}も改変したのではないかというような時には、タス機関の記録から認識されるA_nの証明書と提出された、即ちハッシした文書が使われて、その後の、問題となっている証明書値c_{n-1}を再計算します。もしその値が正しければD_{n-1}は証明されました。利点としては、証明書値c_{n-1}は、A_{n-2}の証明書値と提出された文書から次の証明書記載逓順値c_{n-2}を再計算して証明されます。というのは、もしc_{n-1}がD_{n-2}を処理番号1330で処理した時のものと同じでなければ、他の文書を改変してc_{n-2}と同じ値を得るようにすることは不可能だからです。

第5図に概述されているもっと一般的な記録逓順の手順では、拡大するシリーズの文書が、

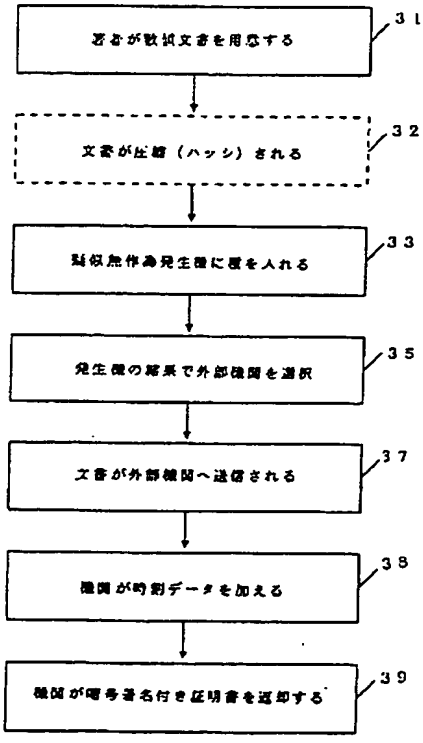
作られる度に、通信の中でまたはタス機関で、処理されます。段階５１では、決定関数でハッシして作られるような、新しい文書の表示が得られ、段階５２では、前の文書を処理して得られた現記録値値と連結されます。段階５３では、この複合表示が処理され、すなわちハッシされ、現在の文書に対する新しい連鎖値を作ります。この値は時間に記録され、証明書にまめられるか、あるいは単に処理表に保存されて段階５４で提示される次の文書に適用されます。以後の処理段階５５、５６はこの文書表示に適用され、この手順は新しい文書が現る度に繰り返されます。



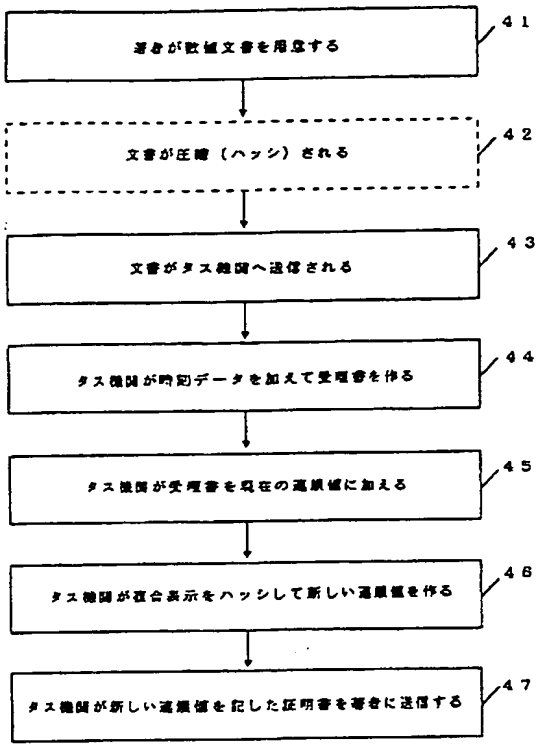
第１図



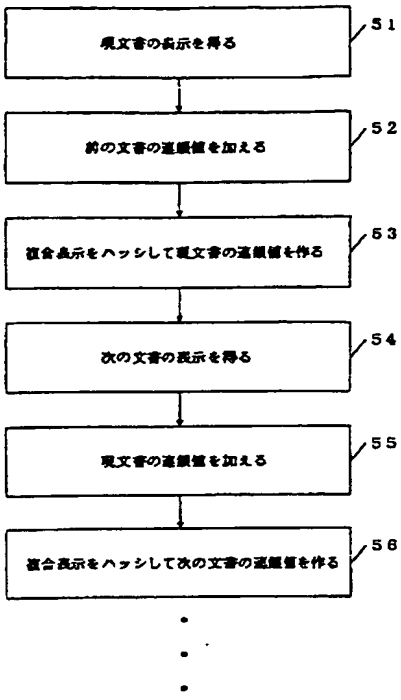
第２図



第３図



第４図



第５図

国 際 調 査 報 告

INT. CL.: H04L 9/00
U.S. CL.: 380/49

U.S. 380/4, 23, 25, 30, 49, 50

Category	Citation of Document	Relevant to Claim No.
A	US, A, 4,145,568 PUBLISHED 20 MARCH 1979, DIFFIE.	1-15
A	US, A, 4,206,315 PUBLISHED 03 JUNE 1980, MEYER ET AL.	1-15

05 OCTOBER 1991
22 NOV 1991
15A/15
PETER PAR. GORDON

by 11/16/91

フロントページの続き

(81)指定国 EP(AT, BE, CH, DE, DK, ES, FR, GB, GR, IT, LU, NL, SE), CA, JP

(72)発明者 ストーネッタ、ウエイクフィールド、スコット、ジュニア
アメリカ合衆国、07960 ニュージャージー州、モリスタウン、ハーディング テラス 34